Wireless Sensor Networks (WSNs) consist of a large number of resource constrained sensor nodes equipped with various sensing devices which can monitor events in the real world. There are various applications such as environmental monitoring, target tracking forest fire detection, etc., which require clock synchronization among the sensor nodes with certain accuracy. However, a major constraint in the design of clock synchronization protocols in WSNs is that sensor nodes of WSNs have limited energy and computing resources. Clock synchronization process in the WSNs is carried out at each sensor node either synchronously, i.e., periodically during the same real-time interval, which we call synchronization phase, or asynchronously, i.e., independently without worrying about what other nodes are doing for clock synchronization. A disadvantage of asynchronous clock synchronization protocols is that they require the sensor nodes to remain awake all the time. Therefore, they cannot be integrated with any sleep-wakeup scheduling scheme of sensor nodes, which is a major technique to reduce energy consumption in WSNs. On the other hand, synchronous clock synchronization protocols can be easily integrated with the synchronous sleep-wakeup scheduling scheme of sensor nodes, and at the same time, they can provide support to achieve sleep-wakeup scheduling of sensor nodes. Essentially, there are two ways to synchronize the clocks of a WSN, viz. internal clock synchronization and external clock synchronization. The existing approaches to internal clock synchronization in WSNs are mostly hop-by-hop in nature, which is difficult to maintain. There are also many application scenarios where external clock synchronization is the only option to synchronize the clocks of a WSN. Besides, it is also desired that the internal clock synchronization protocol used is fault-tolerant to message loss and node failures. Moreover, when the external source fails or reference node fails, the external clock synchronization protocol should revert back to internal clock synchronization protocol with/without using any reference node. Towards this goal, first we propose three fully distributed synchronous clock synchronization protocols, called Energy Efficient and Fault-tolerant Clock Synchronization (EFCS) protocol, Weighted Average Based Internal Clock Synchronization (WICS) protocol, and Weighted Average Based External Clock Synchronization (WECS) protocol, for WSNs making use of peer-to-peer approach. These three protocols are dynamically interchangeable depending upon the availability of external source or reference nodes. In order to ensure consistency of the synchronization error in the long run, the neighboring nodes need to be synchronized with each other at about the same real time, which requires that the synchronization phases of the neighboring nodes always overlap with each other. To realize this objective, we propose a novel technique of pullback, which ensures that the synchronization phases of the neighboring nodes always overlap. In order to further improve the synchronization accuracy of the EFCS, WICS, and WECS protocol, we have proposed a generic technique which can be applied to any of these protocols, and the improved protocols are referred as IEFCS, IWICS, and IWECS respectively. We then give an argument to show that the synchronization error in the improved protocols is much less than that in the original protocols. We have analyzed these protocols for bounds on synchronization error, and shown that the synchronization error is always upper bounded. We have evaluated the performance of these protocols through simulation and experimental studies, and shown that the synchronization accuracy achieved by these protocols is of the order of a few clock ticks even in very large networks. The proposed protocols make use of estimated drift rate to provide logical time from the physical clock value at any instant and at the same time ensure the monotonicity of logical time even though physical clock is updated at the end of each synchronization phase. We have also proposed an energy aware routing protocol with sleep scheduling, which can be integrated with the proposed clock synchronization protocols to reduce energy consumption in WSNs further.